NONMETRIC MULTIDIMENSIONAL SCALING UNDER SAS

Forrest W. Young, University of North Carolina
Robert M. Hamer, Medical College of Virginia
Rostyslaw Lewyckyj, University of North Carolina

In recent years, multidimensional scaling (MDS) has enjoyed both increasing popularity and rapid development. This comparatively new technique grew out of psychophysical scaling and can be traced back to 1952 (Torgerson, 1952). Torgerson's original treatment of MDS required that the distance judgments be metric data.

Nonmetric MDS, which makes weaker assumptions, was proposed in 1962 (Shepard, 1962), and improved by Kruskal (1964).

An example is in order. Consider the map of the United States (Figure 1), with the locations of ten cities indicated. (All tables and figures are located at the end of this paper.) Using a ruler, and the known scale factor of the map, we may calculate the distances between each city on the map and each other city on the map (Table 1). We may then throw the map away. MDS, in this, its simplest form, is a technique which allows us to reconstruct the coordinate space, and, thus, certain features of the map, from the distances alone. Table 2 contains the SAS statements (and the data) necessary to perform this analysis, using PROC ALSCAL. We can then visualize a map of the outline of the United States around these points (Figure 2) and see how accurately PROC ALSCAL reproduced the map of these cities.

Of course, MDS has uses beyond producing maps, although geographers can use PROC ALSCAL to produce maps from certain types of imperfect data. The history of MDS is a history of rapidly expanding methods and uses.

As stated earlier, MDS made its first published appearance in 1952 and nonmetric MDS in 1962. The next milestone was the ability to simultaneously scale multiple sets of judgments about the same set of stimuli, producing both a group space (in which the stimuli are located) and a subject space (showing the weight which the subjects attribute to each dimension). This form of MDS, called Individual Differences Scaling, was first incorporated in the computer program, INDSCAL (Carroll and Chang, 1970) an acronym for Individual Differences SCALing. Several other researchers worked on the same or similar problems simultaneously, but Carroll and Chang produced the first workable computer program. INDSCAL performs a metric analysis.

The next important development was ALSCAL (Takane, Young, and de Leeuw, 1977), the first nonmetric individual differences procedure. ALSCAL used an entirely new numerical procedure for optimizing the relationship between the data and the squared distances, and was much faster than previous programs. The numerical method used was termed "Alternating Least Squares" (ALS), known in some numerical analysis literature as a "successive block algorithm". This type of algorithm is based on the principle that if a set of parameters to be estimated is divided up into mutually exclusive subsets, and there exist conditional least squares estimates for each set (conditioned on fixed values for the parameters in the other sets), then successively estimating parameters in each set while regarding those in other sets as fixed will produce convergence on the locally least squares estimates for all sets.

Since 1977, work has been expanded and accelerated. The work of Young (1975) on an individual differences asymmetric procedure which uses a set of stimulus weights to fit asymmetric data to an asymmetric model, has been incorporated. The unfolding model (Coombs, 1964) has been incorporated. Improvements discussed in Young, Takane, and Lewyckyj (1978) has been incorporated. Thus, the version of PROC ALSCAL discussed in this paper is able to fit a very general model which subsumes most models discussed above, and many others.

PROC ALSCAL is capable of fitting this wide variety of models to a wide variety of data. Data may be two or three way, rectangular or square, symmetric or asymmetric, conditional or unconditional, discrete or continuous, replicated or unreplicated. PROC ALSCAL recognizes and handles missing data. The observed data may be measured at levels ranging from binary to ratio. The total amount of data which PROC ALSCAL can handle is limited only by the amount of storage available to SAS.

The simplest model which PROC ALSCAL will fit is the classical MDS model (the simple euclidean model) Torgerson (1952) used in his formulation. The model is

\[ d_{ij}^2 = \sum_{a=1}^{r} (x_{ia} - x_{ja})^2, \]

where \( d_{ij}^2 \) is the squared distance between points \( i \) and \( j \), \( r \) is the dimensionality of the model space, \( x_{ia} \) is the projection of stimulus \( x_i \) on dimension of \( a \), and \( x_{ja} \) is the projection of stimulus \( x_j \) on dimension \( a \). The model requires that the data be square and symmetric but they can have any of the other aspects discussed above. Most of the early nonmetric MDS programs (the so-called 'Shepard-Kruskal' method) fit this model to ordinal data.

Coombs' (1964) unfolding model results from using two sets of stimuli (the "X" set and the "Y" set) and modeling only the distances between elements of different sets.
This model is
\[ d_{ij}^2 = \sum_{a=1}^{r} (x_{ia} - y_{ja})^2 \]
where \( d_{ij}^2 \) is the squared distance between points \( x_i \) and \( y_j \), \( r \) is the dimensionality of the model space, \( x_{ia} \) is the projection of row stimulus \( x_i \) on dimension \( a \), and \( y_{ja} \) is the projection of column stimulus \( y_j \) on dimension \( a \). Due to the nature of the model, the data matrix must be rectangular, but can have any of the other aspects discussed above.

PROC ALSCAL also fits the weighted individual differences model, the model which INDSCAL fits. This model is
\[ d_{ijk}^2 = \sum_{a=1}^{r} w_{ka} (x_{ia} - y_{ja})^2 \]
where \( d_{ijk}^2 \) is the squared distance between stimuli \( x_i \) and \( x_j \), \( x_{ia} \) is the projection of stimulus \( x_i \) on dimension \( a \), \( y_{ja} \) is the projection of stimulus \( y_j \) on dimension \( a \), and \( w_{ka} \) is the weight for subject \( k \) on dimension \( a \). This model allows a common space for the stimuli, and each subject, via a set of weights, to expand or contract the space individually along each dimension. PROC ALSCAL can fit the model to metric or nonmetric three-way square data.

There exists a corresponding weighted unfolding model. The model is
\[ d_{ijk}^2 = \sum_{a=1}^{r} w_{ka} (x_{ia} - y_{ja})^2 \]
where all symbols have exactly the same meanings as they have previously. This is an asymmetric individual differences unfolding model, with a weight for each subject to use on each dimension, and a weight for each row stimulus on each dimension. The data must be three-way and asymmetric.

When fitting all of the models described, normalizations take place after parameters are estimated. Depending on the model fit and the situation involved, it may make sense to normalize within certain subsets of the data. If normalization across the entire set of data is permitted, the data are said to be unconditional. If normalization only within each individual or replicate (matrix) is permitted, the data are said to be matrix conditional. If normalization within each row of each matrix only is permitted, the data are said to be row conditional. Of course, if there is only one matrix, unconditional and matrix conditional have the same effect.

The next section outlines some of the major options and parameters in PROC ALSCAL. The PROC ALSCAL instructions (SAS Technical Report S-113, Young and Lewyckyj, 1979) covers these options and parameters in greater detail. The options and parameters to PROC ALSCAL may be divided into four classes: data, analysis, model, and algorithmic.

Data Options and Parameters

**DATA**
- dataset name. This parameter allows the user to specify the SAS dataset name containing the data to be used by PROC ALSCAL. PROC ALSCAL expects to find one or more similarity or dissimilarity matrices in this dataset.

**SHAPE**
- There are three possible shapes for the data: SYMMETR for symmetric data; ASYMMETR for asymmetric data; RECTANGU for rectangular data.
ROWS = The number of rows is needed only when SHAPE = RECTANG, in which case it specifies the number of rows in the matrix.

LEVEL = This parameter specifies the transformation to be used on the data, and may be NOMINAL, ORDINAL, INTERVAL, or RATIO.

CONDITIN = This parameter is used to specify the conditionality of the data, and may be UN (for unconditional), MATRIX, or ROW.

Model options and parameters

MODEL = This parameter is used to specify the model. Since there are four possible models (ignoring the unfolding models) this parameter has four possible values: EUCLID (Euclidean unweighted model), INDSCAL (Euclidean with "subject" weights), ASYMSCAL (Euclidean with "stimulus" weights), and ASYMINDS (Euclidean with both sets of weights).

Algorithmic options and parameters

There are a variety of options and parameters which control details of the algorithm, none of which will be mentioned here.

I/O options and parameters

A variety of options and parameters are available which allow the user to read in initial values for the weights and configurations from a SAS dataset and to force these to remain fixed for the remainder of the analysis. There are also options and parameters to allow the user to place various coordinates and weights into a SAS dataset, suitable for manipulation by other SAS procedures or by later invocations of PROC ALSCAL.

The first example has been discussed already. It consists of an analysis of ratio intercity distances.

The next example is based on ranked driving distances among the same set of cities as the previous example. Please note that the ranks themselves have been used as the data for this analysis. Thus, all the metric information in the distances has been concealed from PROC ALSCAL. The results are interesting. Except for a slight rotation (all "classical" MDS solutions are only determined up to a rotation) the results are virtually identical to the metric analysis. Thus, PROC ALSCAL has recovered the metric information from the nonmetric data (Figure 3).

The final example is that of fitting asymmetric data to an asymmetric model. The data used were based upon volumes of export from each country in the set of each other country in the set. Since two countries rarely export exactly the same amount to another country as they receive from it, the model to be fit needs to be asymmetric. Since there was only one data matrix, there are no subject weights in the model. There are, however, stimulus weights. These stimulus weights are what makes the model asymmetric (Figure 4).

References


Table 1
Distances Among Ten U.S. Cities

Table 2
SAS Statements and Data

Figure 1
Ten U.S. Cities

Figure 2
Ten U.S. Cities
Two Dimensional MDS Solution
Figure 3
Ranked Driving Analysis

Figure 4
Asymmetric Analysis of Asymmetric Data